

March 2, 1970

QUARTERLY PROGRESS REPORT
CONVOLUTIONAL CODING TECHNIQUES
FOR DATA PROTECTION

NASA GRANT NGL-15-004-026

Submitted to: Flight Data Systems Branch
NASA Goddard Space Flight Center
Greenbelt, Md. 20771
ATT: Dr. Robert W. Rocheille (Code 710)

and: National Aeronautics and Space Administration
Washington, D. C. 20546
ATT: Miss Winnie M. Morgan (Code US1)
Technical Reports Assistant

Principal Investigator: Dr. James L. Massey
Professor of Electrical Engineering
University of Notre Dame
Notre Dame, Ind. 46556

Research Assistants: J. Chang
J. Geist
W. Hartman
G. Seguin

RESEARCH PERIOD REPORTED: November 16, 1969 to February 15, 1970

**CASE FILE
COPY**

I. Research on Sequential Decoding

The last quarterly progress report included mention of the development of a modified Fano sequential decoding algorithm by J. Geist in which the threshold spacing parameter Δ is eliminated. A flowchart for this algorithm is given in Fig. 1, and the algorithm will now be explained in reference to this flowchart.

The new quantity introduced by Geist is the "forward blocking value" which is denoted K in Fig. 1. This quantity is made to contain the largest value of all the node values on nodes which are blocking any further forward moves, i. e. for which the test " $V + b_F \geq T$?" gives a negative result. T , of course, represents the threshold. As in the original Fano algorithm, the threshold is lowered only when blocking occurs on a backward move, i. e. when the test " $V - b_B \geq T$?" is answered in the negative. The heart of the Geist algorithm is that the threshold is then lowered to the maximum of K and $V - b_B$, i. e. to the largest value just sufficient to permit a new move (either forward or backward) to be made. Some economy of computation is obtained by noting that when $V - b_B$ (the backward blocking value) wins the comparison, there is no point in moving forward again after reducing the threshold T to $V - b_B$ since this will not be enough to get by any of the forward blocking nodes. Hence, a backward move is immediately made in this case--in contrast to the ordinary Fano algorithm where a forward move always follows decrementing of the threshold.

As in the original Fano algorithm, the threshold is only raised or "tightened" upon a first visit to a node. In the Geist algorithm, the tightening is done by changing the threshold to the value of the new node encountered. With a little reflection, it can be shown that the "first

visit" test can be implemented by checking whether the node from which the forward move was made was itself precisely at threshold. If so, the arrival is a "first visit" for which the threshold must be tightened.

Conceptually, the Geist algorithm fills a plaguing void in that it shows how the parameter Δ which many investigators have felt to be inessential may be eliminated. The original Fano algorithm can then be considered as a "quantized" version of the Geist algorithm since it has the effect of reducing node values to equivalent values in each quantization bin. Jelinek had earlier given both a quantized and an unquantized form of his stack algorithm, and it had been puzzling that no unquantized Fano algorithm had been found--especially since the unquantized Jelinek algorithm is more fundamental and easy to understand than its quantized counterpart.

There are two noteworthy aspects of the Geist unquantized-Fano algorithm: (1) The threshold is always lowered to precisely that value which permits exploration of a new part of the value tree, i. e. there is never a decrease insufficient to open up any new nodes nor a drastic decrease which opens up more than the most favorable new part of the tree--both of which can occur in the quantized Fano algorithm. (2) There is no need to make an analytical or experimental determination of an optimum threshold spacing parameter Δ . The first of these facts indicates that the unquantized algorithm will give optimum error probability performance--in fact the limiting performance as Δ approaches 0 in the usual Fano algorithm. For channel decoding purposes, this is not of much practical importance as a good computational value of Δ usually

is achieved with little sacrifice in error probability. It may however be of interest in other applications of sequential decoding (e. g. in sub-optimal but practical solutions of dynamic programming tasks where, moreover, there is usually no way to determine a good quantizing value.) Fact (1) also suggests that the unquantized algorithm may reduce the number of computations compared to the quantized algorithm. If so, this would be of much practical interest since the effort in a single computation is nearly identical for both forms of the algorithm--in sharp contrast to the Jelinek algorithm where the unquantized computations are considerably more time-consuming. Extensive simulations on the binary symmetric channel indicated that the unquantized algorithm does indeed often reduce the computation to decode a frame when the noise is severe. This might be expected since in this case there are many paths near to tying for the best path and the unquantized algorithm prevents many false starts. The gain in most cases, however, is quite small and is more than lost by the fact that the quantized algorithm finishes much more quickly when the noise is light, i. e. when a bit more daring in the tree search pays off. Our conclusion is that the unquantized algorithm offers no practical advantage for decoding on the binary symmetric channel.

The above results were presented orally in the following reference:
J. M. Geist, and J. L. Massey, "A Comparison of the Fano and Jelinek Sequential Decoding Algorithms," (abstract only) presented at the First NASA Coded Communications Conference, JPL-Pasadena, Calif., Feb. 27, 1970.

Mr. Geist has now completed his research requirements for the Ph. D. degree as a research assistant under this grant and since February has been on active naval duty as an instructor at the U. S. Naval Postgraduate School

in Monterey, Calif. He is presently writing his dissertation which will include considerable detail on the above algorithm and will be distributed as a technical report under this grant sometime about next May.

Geist has also completed detailed comparative simulations of the usual Fano algorithm and both forms of the Jelinek algorithm. These simulations indicate that in almost all cases a time advantage of about 20 to 50 per cent can be obtained with the quantized Jelinek algorithm compared to the usual Fano algorithm. Details of this work are contained in the following report which is currently in final typing status and will soon be distributed:

J. M. Geist, "A Comparison of the Fano and Jelinek Sequential Decoding Algorithms," Tech. Report EE-701, Dept. of Elec. Engr., Univ. of Notre Dame, Notre Dame, Ind. 46556, Jan. 31, 1970.

Plans for next quarter:

During the next quarter, the Geist algorithm will be simulated on a 3 bit quantized Gaussian noise channel to determine whether it offers any computational advantage compared to the usual Fano and Jelinek algorithms. With the many different possible metric increments for such a channel, it is conceivable that the Geist algorithm may be a practical scheme for such a channel in contrast to the rate $1/2$ binary symmetric channel cases for which simulations were performed earlier. It is also intended to inquire further into other applications of sequential decoding where the Geist algorithm may have promise.

II. Research on Good Convolutional Codes

L. Bahl and F. Jelinek (Private Communication) have recently devised a class of so called "complementary rate $1/2$ non-systematic codes" which appear very promising because of their large free distance. At the following presentation:

J. L. Massey, "Non-Systematic Convolutional Codes for Sequential Decoding," (abstract only) First NASA Coded Communications Conference, JPL-Pasadena, Calif., Feb. 27, 1970.

the principle investigator showed that these codes could be encoded very simply by the same type of trick used to encode the "quick look-in" codes described in the quarterly progress report of June 5, 1969. The Bahl-Jelinek codes have the property that the two generator polynomials are related by

$$G_2(D) = G_1(D) + D + D^2 + \dots + D^{M-1}$$

where M is the degree of the generators. Letting

$$G_1(D) = 1 + g_1 D + g_2 D^2 + \dots + g_{M-1} D^{M-1} + D^M$$

it is readily seen that the circuit in Fig. 2 is a valid encoder for this code and uses essentially only the number of modulo-two adders that would be needed to implement $G_1(D)$ alone.

Plans for next quarter:

During the next quarter, it is planned to make an intensive effort to find good time-varying convolutional codes. Theoretical considerations indicate the possibility that these codes may be far superior to the fixed codes that have always been used in practice. In particular, a class of at least relatively-simply encoded codes of the time-varying class studied on an ensemble basis several years ago by the principal

investigator who showed that this restricted ensemble had all the desirable properties of the full ensemble will be resurrected in the hope that the relatively simply structure will made the search for good codes of this type feasible.

III. Research on Sub-Baud Coding

An extensive investigation is being conducted by G. Seguin under this grant in order to find good sub-baud codes, i. e. for the selection of n bit sequences to control the polarity of the modulated envelope in each $1/n$ th of the basic baud waveform. The aim of such coding is, of course, bandwidth spreading to reduce multipath and--in many user systems--multi-user interference. The partial results below apply to reducing multipath interference in a single user system and give promise of leading to some tangible rewards.

Let $\underline{a} = (a_1, a_2, \dots, a_n)$ be an n bit sequence. Let $f_i(\underline{a})$ denote the vector made up of the first i bits of $\underline{a}$, and let $b_i(\underline{a})$ denote the last i bits of $\underline{a}$. The finite autocorrelation function for $\underline{a}$ is the function $F(k)$ defined as

$$F(k) = k - 2 d_H(f_k(\underline{a}), b_k(\underline{a}))$$

where $d_H(,)$ denotes the Hamming distance between the two sequences.

In a continuous digital signalling scheme, it is desirable to reduce the off-peak (i. e. $0 < k < n$) values of the even and odd correlation functions, $E(k)$ and $\theta(k)$ respectively, to the smallest possible magnitudes since these measure the possible contributions of multipath to a correlation receiver's output. In terms of $F(k)$, these correlation functions may be written as

$$E(k) = F(k) + F(n-k)$$

$$\theta(k) = F(k) - F(n-k).$$

The following approach to finding a good sequence a has been developed by G. Seguin. Seguin first placed the constraint that

$$\text{abs}[E(k) - \theta(k)] \leq 2 \quad \text{all } k$$

which ensures that if one of the correlation functions is good then so is the other. The next step is to find a sequence a satisfying this constraint that is also blessed with a good correlation function.

Seguin has shown first that for every n there is at least one sequence satisfying the constraint so that the search for a good sequence is at least not over a void set. Next, Seguin has written a program for the Univac 1107 computer to find all such sequences with $n \leq 31$. The result showed many good sequences at all lengths. It is interesting to note that a Barker sequence must satisfy the above constraint so that the algorithm finds the Barker sequences wherever they exist. There were of course no Barker sequences found for $13 < n \leq 31$.

Plans for next quarter:

This research will be continued actively in the next quarter. The intriguing fact of Seguin's constraint is that it is weak enough to guarantee that sequences can be found to satisfy it while strong enough to obtain the desired simultaneous minimization of the even and odd correlation functions. It is believed that this work can lead to significant new results in sub-baud coding. A preliminary report will be prepared at an early date on the work performed thus far.

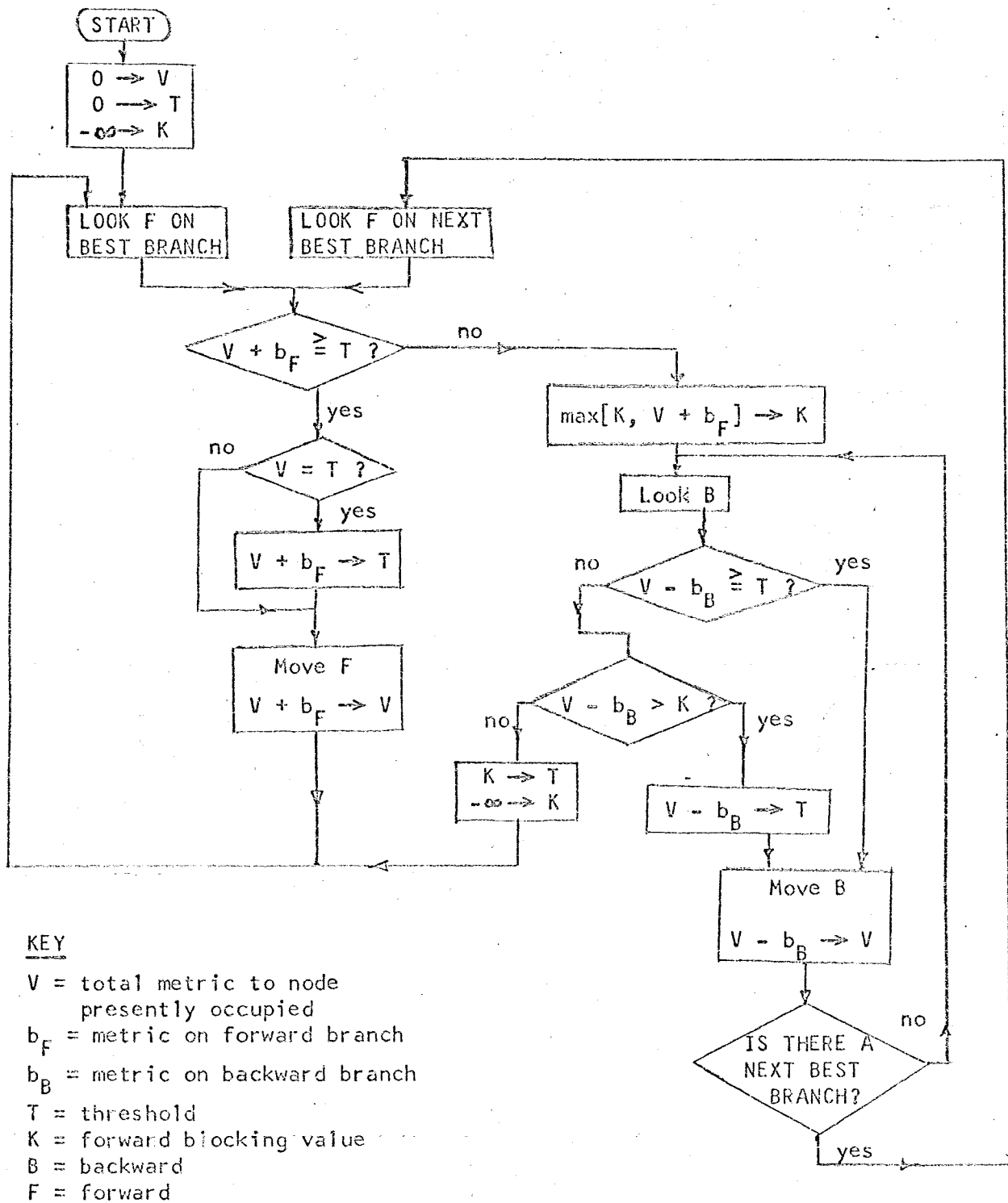
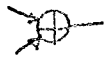


Fig. 1 Unquantized Fano Algorithm (Geist Algorithm)

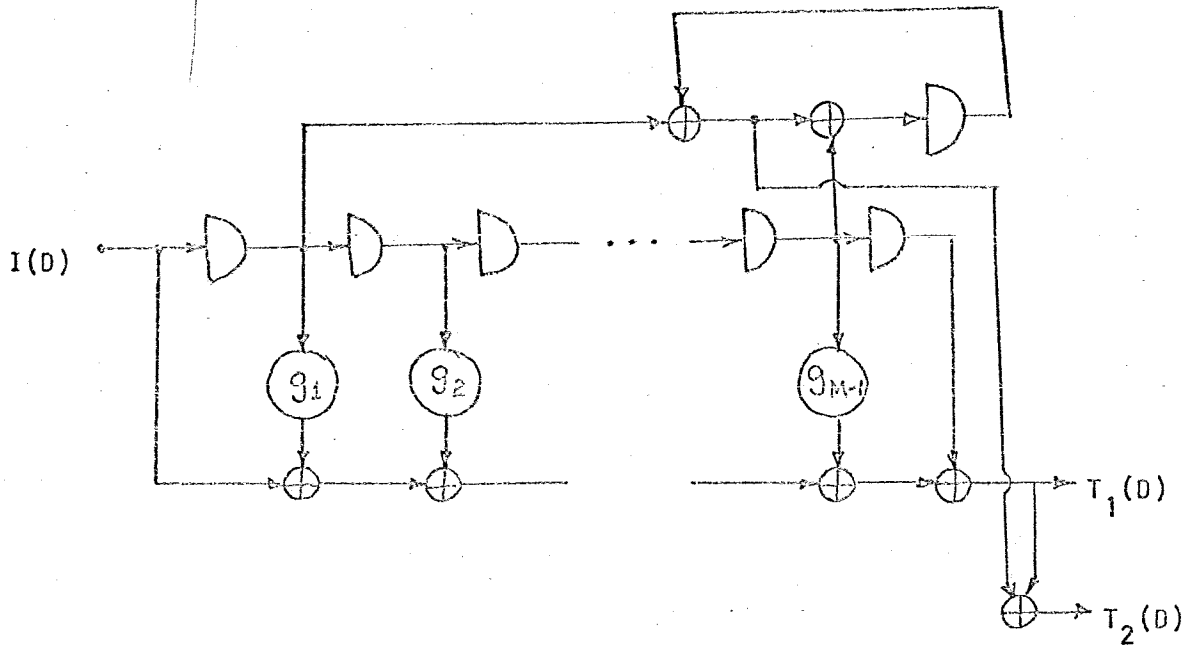
 unit delay

 mod-two adder

Key

$$G_1(D) = 1 + g_1 D + g_2 D^2 + \dots + g_{M-1} D^{M-1} + D^M$$

$$G_2(D) = G_1(D) + D + D^2 + \dots + D^{M-1}$$



$$T_1(D) = I(D) G_1(D)$$

$$T_2(D) = I(D) G_2(D)$$

Fig. 2 Simple Encoder for the Bahl-Jelinek
R = 1/2 Non-Systematic Codes